



# 自助关联分析方案

润乾软件出品



# 自助关联分析是什么？



多维分析通常针对单个表（CUBE），但实际业务常常会涉及多表关联。那么如何让多维分析支持有JOIN关联的情况？

**宽表** 将多表事先关联好形成逻辑或物理单表

**自动关联** 用户选择字段后，自动完成关联。

**由用户指定关联** 将表间关系暴露给用户，让用户自行关联。





## 常规关联分析的弊端

**宽表**——维度层次多，造成冗余（逻辑数量可多至无穷），经常重新建模 ❌

**自动关联**——仅适用于简单的情况，遇到同维字段或自关联时不能正确匹配 ❌

**由用户指定关联**——面对复杂的表间关系会使业务用户比较“晕” ❌

润乾报表多维分析基于DQL Server，首次解决业务人员实施多表关联自助分析的难题。

**DQL (Dimensional Query Language, 一种以维度为核心的类似SQL查询语言)** ——仅需一次性建模，将数据表的结构及表间关联关系保存于元数据文件中。有新关联查询需求时无需更新模型，只有数据库中表及字段以及关联发生了改变才要再次改造模型。





1

不需要逻辑和物理宽表，临时根据关联需要生成 join，应用灵活查询需求

2

通过将表看做维度、外键指向表的字段看做维度的属性后，页面可通过树状目录的方式逐级展开，避免了自动关联和用户关联的弊端



# 目录

CONTENT

01

方案概述

02

基本外键关联

03

多层分析

04

同维关联

05

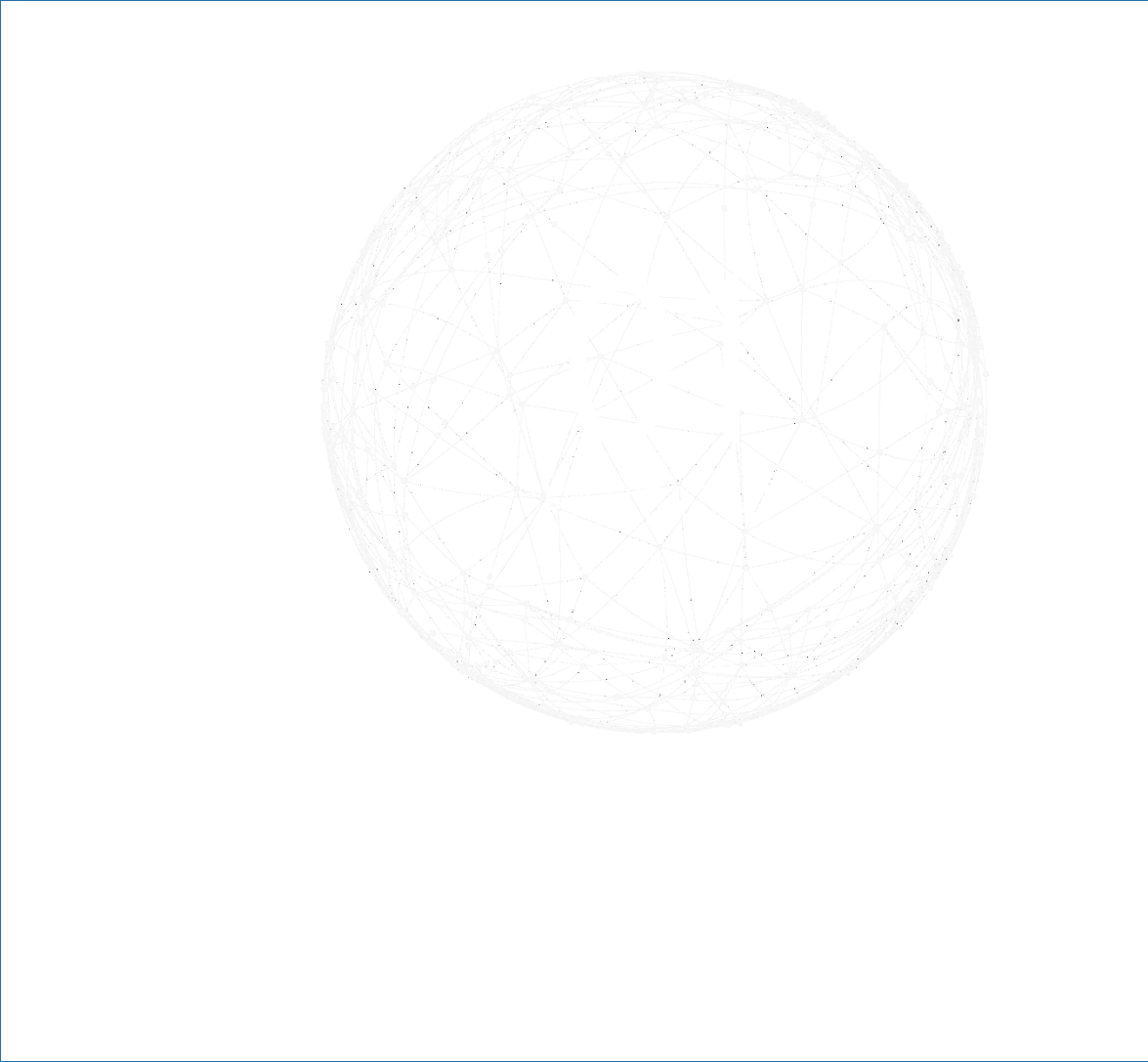
汇总表

06

大数据跨库分析

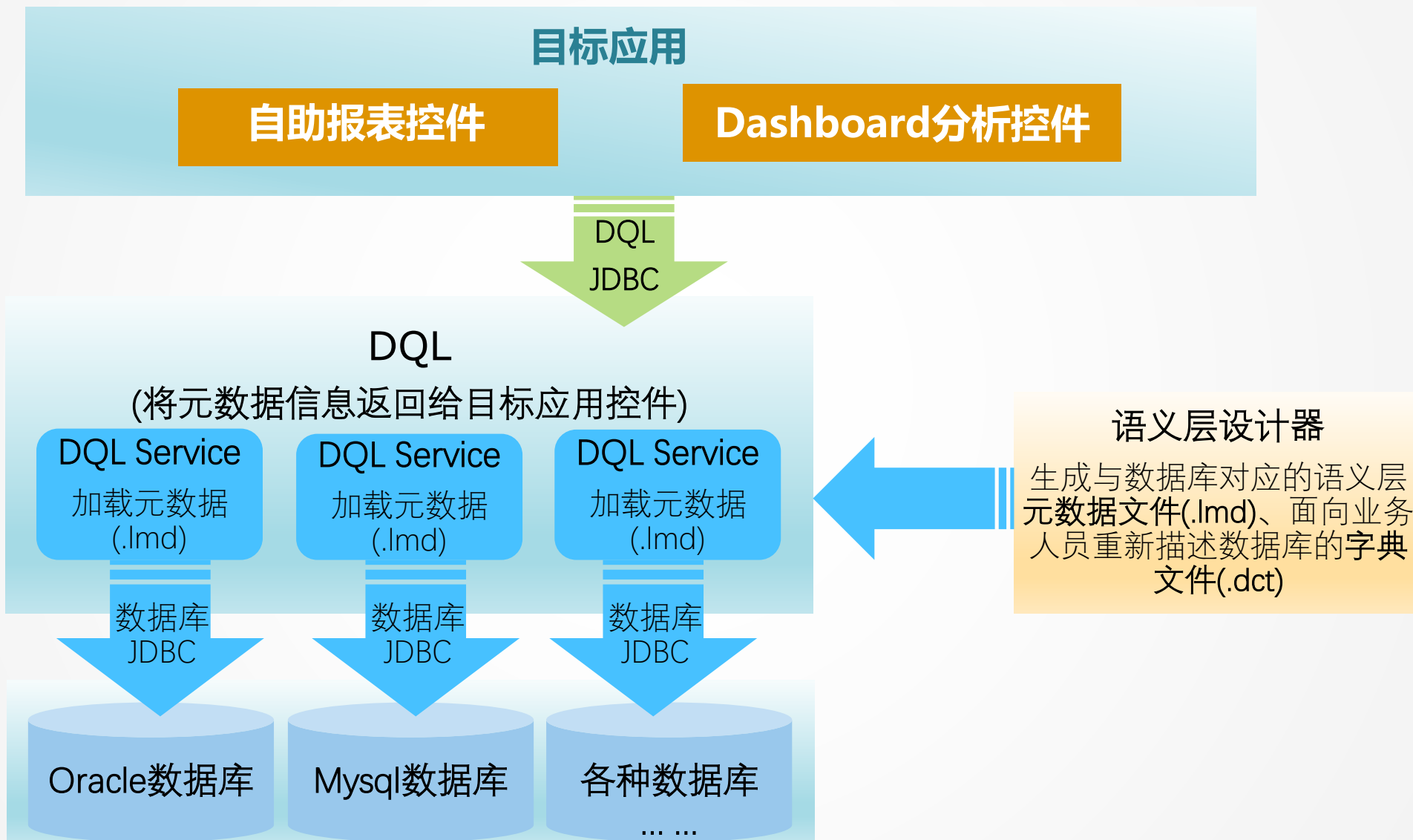
07

Dashboard分析





# 应用方案结构图





## 外键属性化

- 表中字段（可能多个）与另一个表（可能是本身）主键关联，构成外键的字段称为外键字段。
- 外键指向记录可看作本表子属性引用。外键属性化可以使多表关联查询转化成单表查询。并且允许多层和递归引用。

## 同维表等同化

- 主键一一一对应关联的表互为同维表。
- 同维表之间的关系是对等的，从任何一个表都可以直接引用到其它同维表的字段。

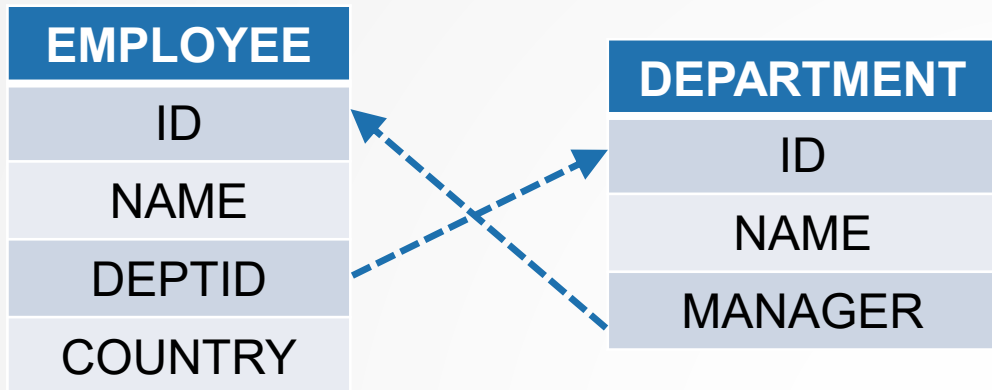
## 按维对齐连接

- 多表连接时针对每个表独立设定汇总维度，无须关心表间连接关系，降低问题难度。





# 外键属性化



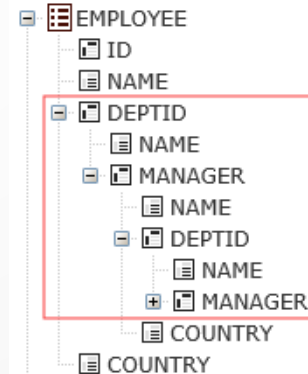
针对这种很常规的表结构设计。  
如何获取哪些美国籍员工有一个中国籍经理呢？

```
SQL: SELECT A.* FROM EMPLOYEE A JOIN DEPARTMENT B ON  
A.DEPTID=B.ID JOIN EMPLOYEE C ON B.MANAGER=C.ID WHERE  
A.COUNTRY='美国' AND C.COUNTRY='中国'
```

//SQL句子较长，由于  
EMPLOYEE表需要两次参与JOIN  
，还需要为它起个别名加以区分

```
DQL: SELECT * FROM EMPLOYEE WHERE COUNTRY='美国' AND  
DEPTID.MANAGER.COUNTRY='中国'
```

//外键字段可看做成一个对象，外  
键表的字段被理解为外键字段的  
属性，允许多层递归引用





# 同维表等同化

客户	VIP客户
客户ID	客户ID
客户名称	VIP级别
联系人姓名	折扣率
联系人职务	

在物理数据库的设计中，基于性能、存储空间等的考虑，往往将同一个对象的各种属性，分别存在不同的物理表中。

如：**客户表**和**VIP客户表**。

```
SQL: SELECT A.客户ID,A.客户名称,B.VIP级别 FROM 客户 A LEFT JOIN VIP客户 B ON A.客户ID=B.客户ID
```

//SQL语法需要将两个同维表先连接，再做字段查询

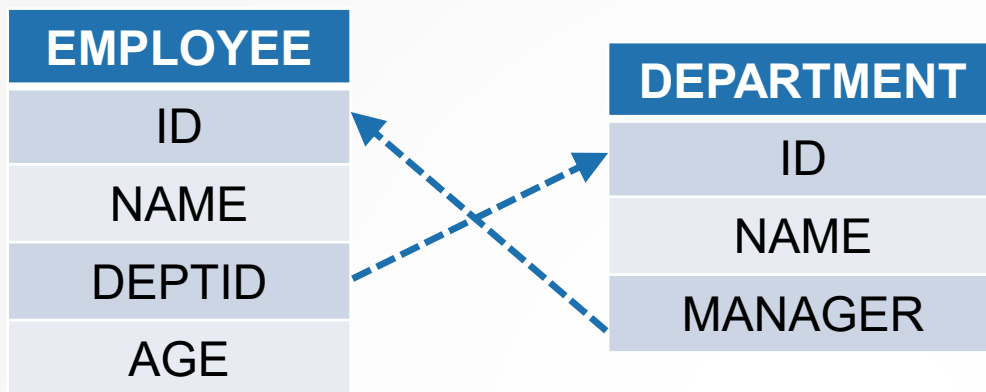
```
DQL: SELECT 客户ID,客户名称,VIP级别 FROM 客户
```

//DQL语法可以只从一个表引用其它同维表字段:

同维表（主键相同的表）可视为一个表，访问其中任何一个均可直接引用其它表的字段。横向分表对上层透明，根据引用字段即时决定连接表，比事先预定义的视图效率更高。



# 按维对齐连接



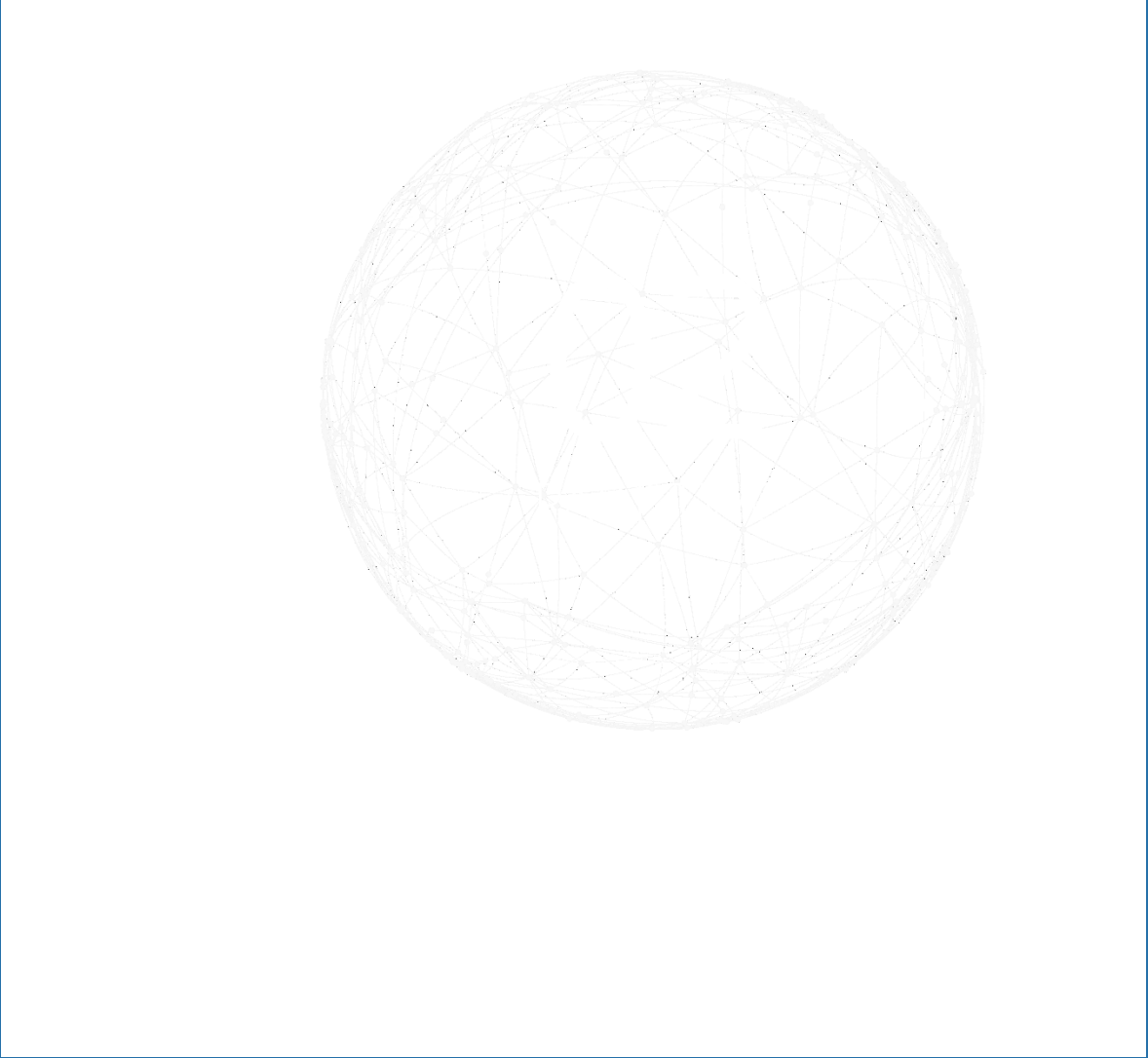
多表连接时针对每个表独立设定汇总维度，无须关心表间连接关系。

```
SQL: SELECT DEPARTMENT.NAME,AVG(EMPLOYEE.AGE) FROM
DEPARTMENT,EMPLOYEE WHERE
EMPLOYEE.DEPTID=DEPARTMENT.ID GROUP BY
DEPARTMENT.NAME
```

//统计部门员工平均年龄，SQL语法需要写清两表之间的关联关系

```
DQL: SELECT DEPTID.NAME,EMPLOYEE.AVG(AGE) FROM
EMPLOYEE BY DEPTID
```

//DQL语法只要各自向目标汇总维度对齐，无须书写表间关系条件，主键作为汇总维度可省略



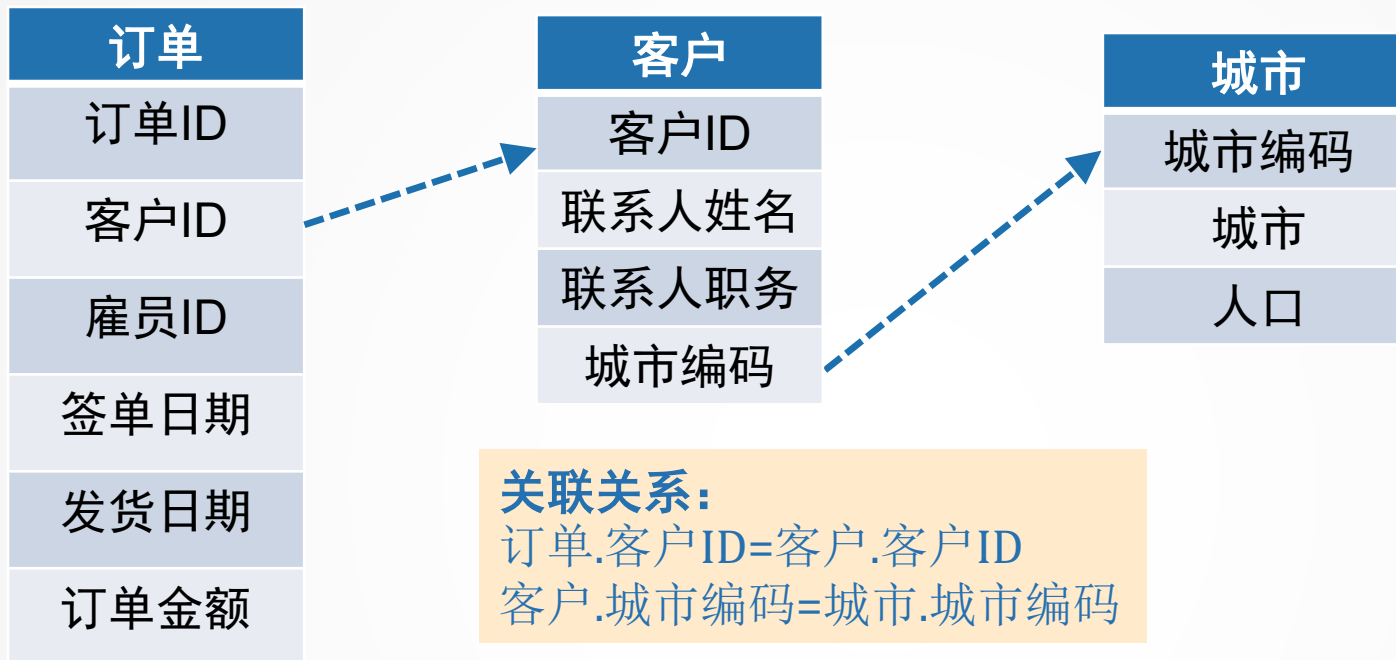


## 语义层

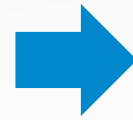
元数据 (.lmd) — 描述数据库中，各个表中的数据结构以及各类表间关系

字典 (.dct) — 定义表或字段的别名

可视 (.vsb) — 设定表或字段的可见



从上图可以看到数据库中有订单、客户、城市三张表，订单与客户使用客户ID做关联，客户与城市表使用城市编码做关联。通过这样的表关系，在多维分析中怎样统计各城市中客户的订单信息呢？



### 配置物理数据源

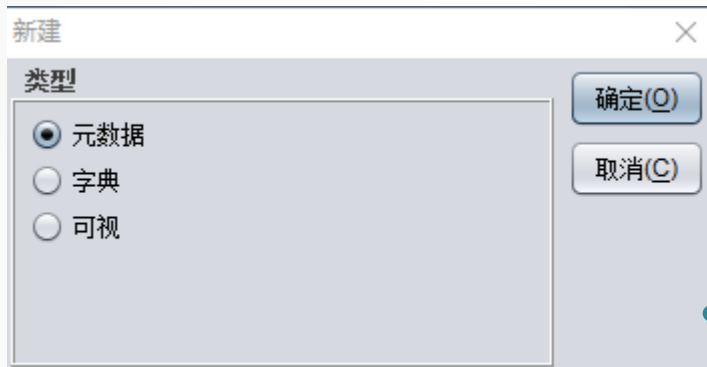
打开元数据层设计器，系统菜单选择【数据源】，配置数据源并连接，连接之前，要保证所连接数据源已启动。

以下演示数据为润乾自带的dqldemo数据库，该数据库不用配置，通过【服务 - 启动示例数据库】菜单可直接启动数据库。

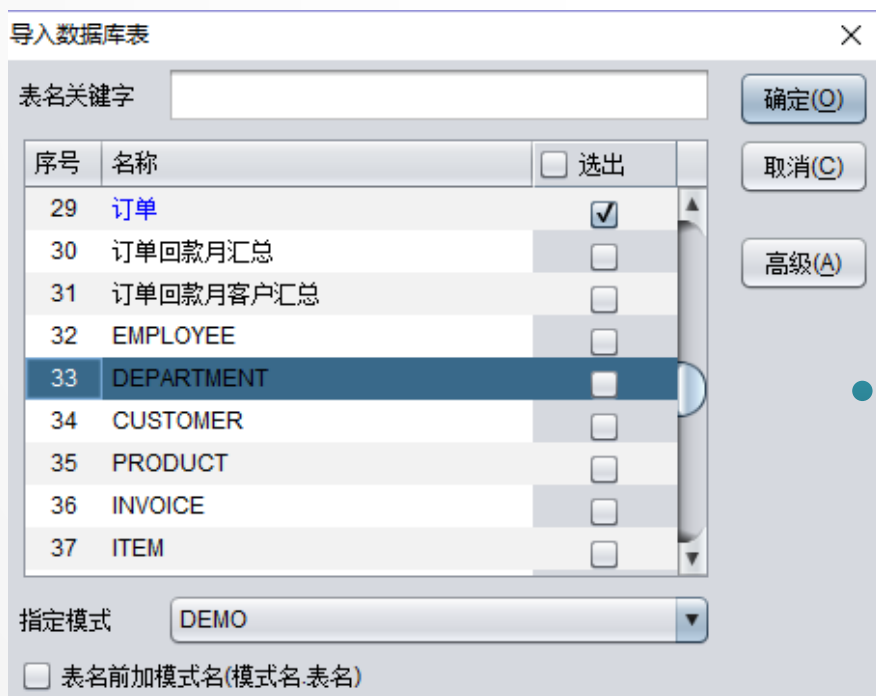




创建元数据文件，文件后缀为.lmd。



文件菜单栏下  
选择【新建】，  
选择元数据



系统菜单下选择【  
导入数据库表】，  
选中订单、客户、  
城市表，导入表结  
构



## 基本外键关联



## 制作元数据



为订单表、客户表、城市表设置主键。

表名: 订单 物理表名: 订单  
记录数: 0 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段  输入字段名关键字进行查找

序号	字段名	主键	数据来源	数据类型
1	订单ID	<input checked="" type="checkbox"/>	订单ID	整数
2	客户ID	<input type="checkbox"/>	客户ID	字符串
3	雇员ID	<input type="checkbox"/>	雇员ID	整数
4	签单日期	<input type="checkbox"/>	签单日期	日期时间
5	发货日期	<input type="checkbox"/>	发货日期	日期时间
6	订单金额	<input type="checkbox"/>	订单金额	双精度数

表名: 客户 物理表名: 客户  
记录数: 0 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段  输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	客户ID	<input checked="" type="checkbox"/>	客户ID	字符串	
2	客户名称	<input type="checkbox"/>	客户名称	字符串	
3	联系人姓名	<input type="checkbox"/>	联系人姓名	字符串	
4	联系人职务	<input type="checkbox"/>	联系人职务	字符串	
5	城市编码	<input type="checkbox"/>	城市编码	整数	

表名: 城市 物理表名: 城市  
记录数: 0 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段  输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	城市编码	<input checked="" type="checkbox"/>	城市编码	整数	
2	城市	<input type="checkbox"/>	城市	字符串	
3	人口	<input type="checkbox"/>	人口	整数	



## 基本外键关联



## 制作元数据

为表字段添加外键，方便通过外键关系引用。

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	客户

外键映射

序号	外键字段	引用字段
1	客户ID	客户ID

为订单表的客  
户字段添加外  
键

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	城市

外键映射

序号	外键字段	引用字段
1	城市编码	城市编码

为产品表中的供  
应商添加外键



## 基本外键关联



## 制作元数据

生成字典文件，设置城市维的显示列名称为城市表的城市名称，文件后缀为.dct。

打开前一步做好的元数据文件，系统菜单下选择【生成字典】

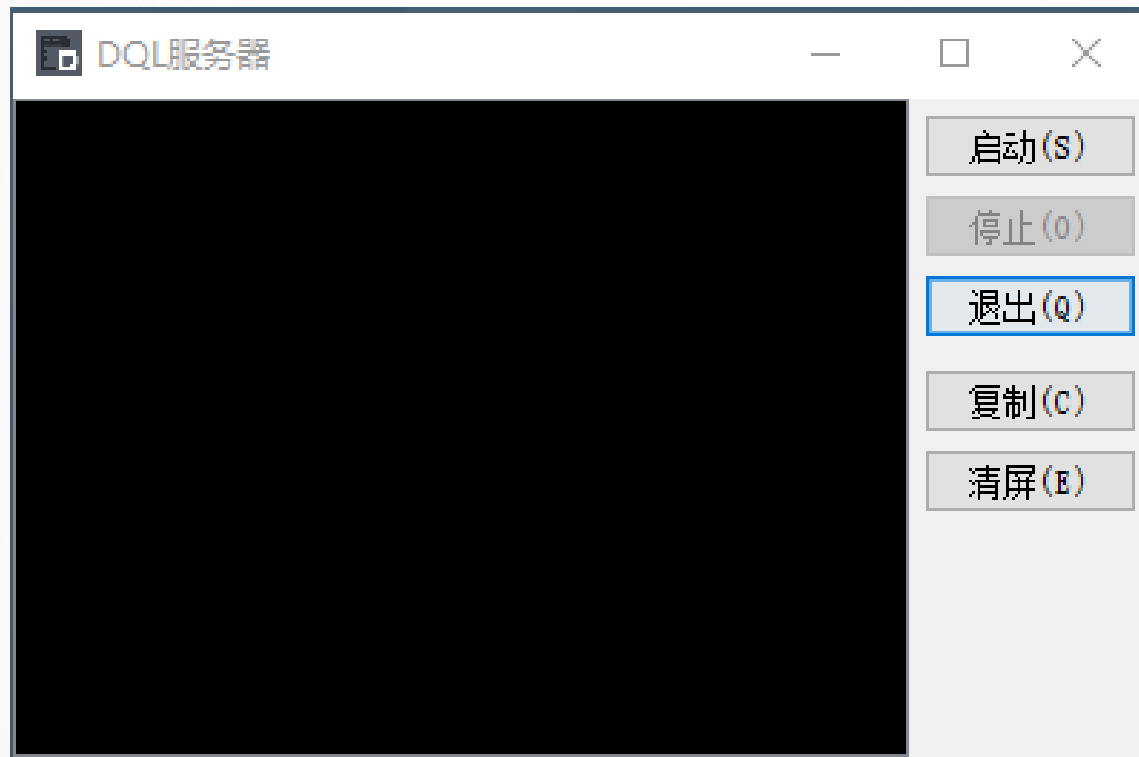
表项	维项	分类项	导入字典			
维项						
序号	维标题	维名称	代码列名称	代码值标题	显示列名称	显示值标题
1	订单_订单ID	订单_订单ID	订单.订单ID			
2	客户_客户ID	客户_客户ID	客户.客户ID			
3	城市	城市_城市编码	城市.城市编码		城市.城市	

设置城市的显示列字段，城市名称就可以用于分析结果

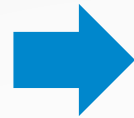
更多关于语义层中元数据和字典文件的细节使用可参考[语义层](#)小节。



需要使用 DQL Server 服务器对元数据文件(.lmd)进行语义解析，DQL 服务器控制台中启动页面如下所示，启动前要保证元数据中使用的物理数据库为启动状态：



- (1) DQL 这个词在界面中出现过多次，想要对DQL做更多了解可参考《[分析教程](#)》；
- (2) 元数据文件需要部署到 DQL Server 服务器，如需了解DQL Server的部署，可以到《[分析教程](#)》中查看关于 DQL 服务器部署集成的专题；
- (3) 字典文件是部署到 web 应用目录下，并在发布分析控件的页面里通过 JS API 去灵活设置，关于这部分的介绍，可以到《[分析教程](#)》JS API 相关主题中查看。



在[WEB应用根目录]/WEB-INF/raqsoftConfig.xml中添加物理数据源和DQL Server数据源

```
<?xml version="1.0" encoding="UTF-8"?><Config Version="1"><Runtime >
//配置应用需要用的数据源
<DBList>
<!--数据使用的物理数据源-->
<DB name="dqldemo">
  <property name="url" value="jdbc:hsqldb:hsqldb://127.0.0.1/reportdemo" />
  <property name="driver" value="org.hsqldb.jdbcDriver" />
<!--数据库类型编号，编号依次代表的数据库类型为：[UNKNOWN:0; ORACLE:1; SQLSVR:2; SYBASE:3; SQLANY:4; INFMIX:5;
FOXPRO:6; ACCESS:7; FOXBAS:8; DB2:9; MYSQL:10; KINGBASE:11; DERBY:12; HSQL:13; TERADATA:14; POSTGRES:15;
DATALOGIC:16; IMPALA:17; ESSBASE:101;] -->
  <property name="type" value="13" />
  <property name="user" value="sa" />
  <property name="password" />
  ... ..
</DB>
<!--DQL Server数据源-->
<DB name="DataLogic">
  <property name="url" value="jdbc:datalogic://127.0.0.1:3366/datalogic" />
  <property name="driver" value="com.datalogic.jdbc.LogicDriver" />
  <property name="type" value="16" />
  <property name="user" value="root" />
  <property name="password" value="root" />
  ... ..
</DB> </DBList> ... ..
</Config>
```



在jsp中定义Tag标签的属性及值，用Tag标签完成多维分析的解析及运算，按用户配置的属性要求将HTML文件流输出到客户端。

Test. jsp:

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
```

```
<%@ taglib uri="/WEB-INF/raqsoftAnalyse.tld" prefix="raqsoft" %>
```

```
<raqsoft:analysev2
```

```
    dataSource="DataLogic"           //指定raqsoftConfig.xml中定义的DQL Server数据源
```

```
    ... .. />
```

```
<script>
```

```
    guideConf.dct="WEB-INF/test.dct"; //指定字典文件路径
```

```
</script>
```

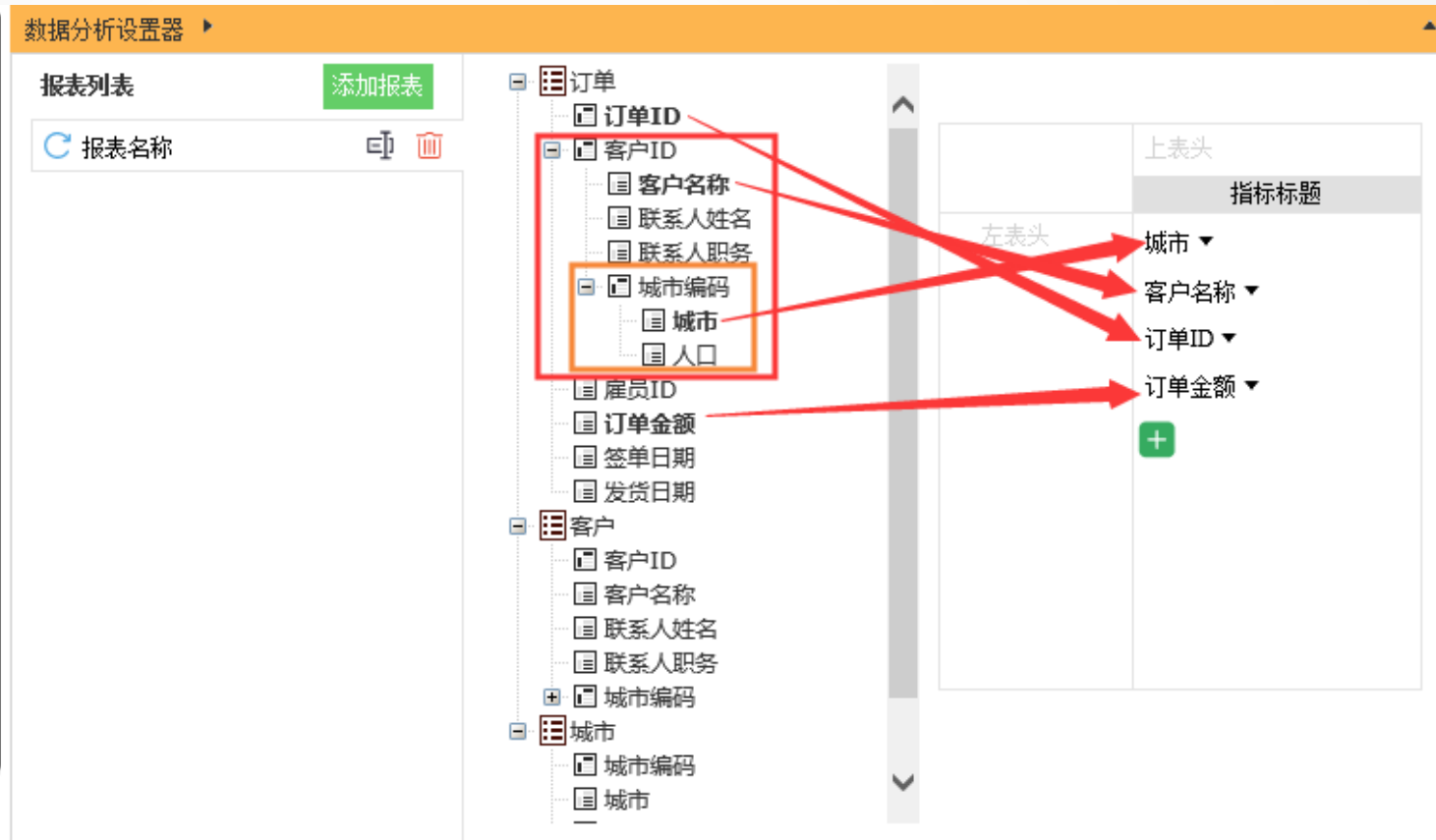




查询订单明细信息：

报表名称

城市	客户名称	订单ID	订单金额
天津	山泰企业	10248	440
青岛	东帝望	10249	1863
南昌	实翼	10250	1813
秦皇岛	千固	10251	670
天津	福星制衣厂股份有限公司	10252	3730
南昌	实翼	10253	1444
天津	浩天旅行社	10254	625
南京	永大企业	10255	2490
深圳	凯诚国际顾问公司	10256	517
深圳	远东开发	10257	1119
深圳	正人资源	10258	2018



元数据中为订单表的**客户ID**和客户表的**城市编码**添加了外键属性，外键关联关系如上图红框所示，在页面上会以树状的形式多层展现。

“客户ID. 城市编码”代表“客户所属城市编码”，而这个字段在客户表中仍然是个外键，那么它的外键表字段可以继续理解为它的属性，也就会有“客户ID. 城市编码. 城市”，即“客户所属城市名称”。



对订单表按城市做分组汇总，汇总每个城市的订单数量和订单总金额：

城市编码	订单ID计数	订单金额求和	订单ID计数-总计	订单金额求和-总计
大连	36	56262	36	56262
上海	9	8701	9	8701
青岛	13	15381	13	15381
南京	83	127462	83	127462
常州	34	58436	34	58436
南昌	14	34098	14	34098
温州	18	52241	18	52241
北京	33	56901	33	56901
天津	222	364937	222	364937
张家口	22	20303	22	20303
石家庄	36	43752	36	43752
秦皇岛	30	42112	30	42112
深圳	142	223525	142	223525
海口	22	25008	22	25008

数据分析设置器

报表列表

添加报表

报表名称

订单

- 订单ID
- 客户ID
  - 客户名称
  - 联系人姓名
  - 联系人职务
  - 城市编码
  - 雇员ID
  - 签单日期
  - 发货日期
  - 订单金额
- 客户
  - 客户ID
  - 客户名称
  - 联系人姓名
  - 联系人职务
  - 城市编码
- 城市
  - 城市编码
  - 城市
  - 人口

城市编码

上表头

指标标题

- 订单ID计数
- 订单金额求和

+

通过在字典中对城市维设置显示字段，这里拖拽产品ID时显示的是城市名称，而不是城市编码。

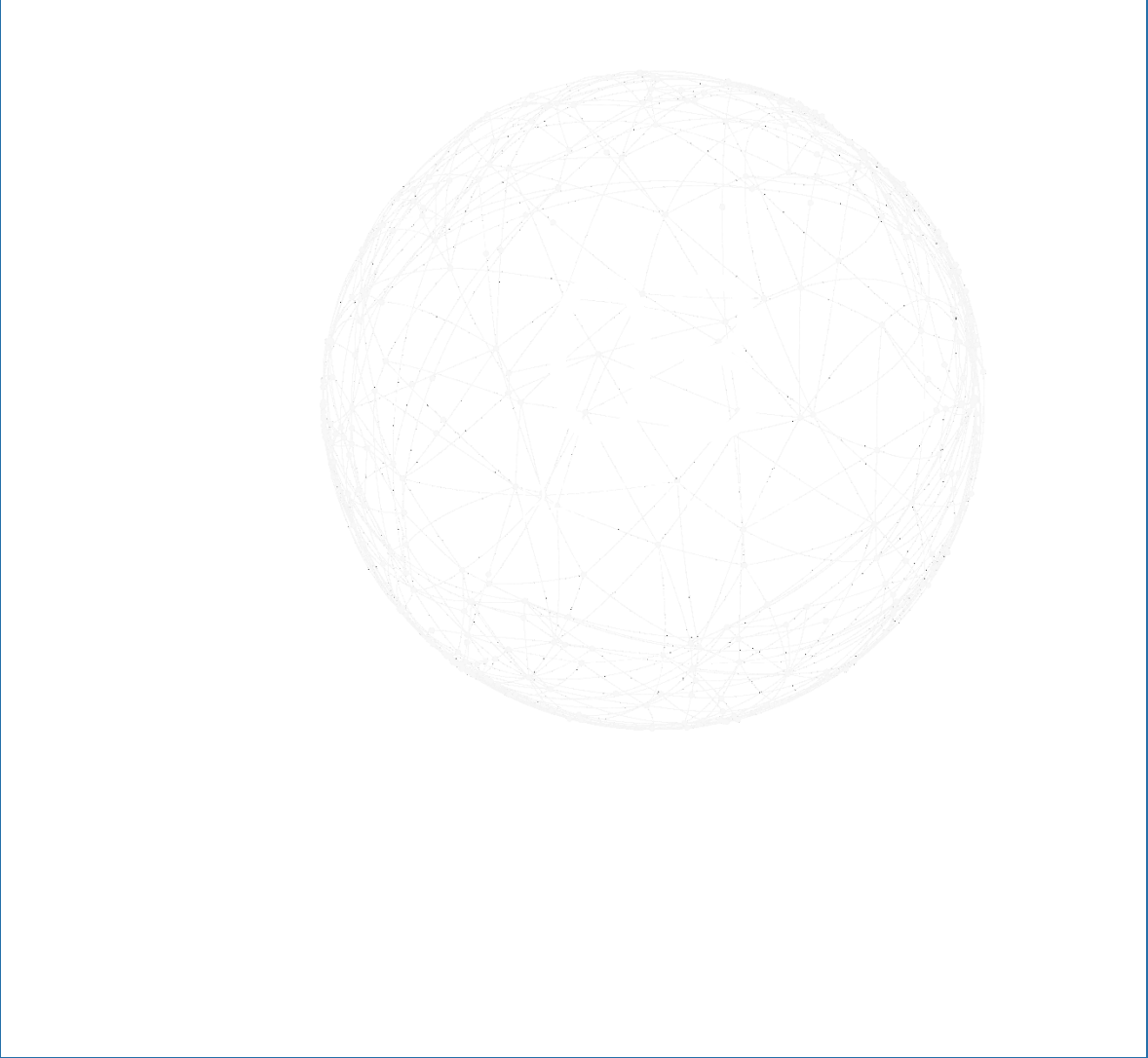




添加过滤条件，统计北京的订单数量与订单总金额：

通过字典中对城市维设置显示字段，这里对城市添加过滤条件时，会自动显示出城市名称。

城市编码	订单ID计数	订单金额求和	订单ID计数-总计	订单金额求和-总计
北京	33	56901	33	56901
总计	33	56901	33	56901





通过常规查询我们学习了如何对订单、客户、城市三张表做基本的明细和汇总分析。但如果业务人员想要按年月日层级来观察不同时期的客户签单情况。该如何实现呢？





打开前面做好的.lmd元数据文件，添加年、月、日三个假表。  
 所谓假表，就是数据库中实际不存在的单字段表。方便用户获得能够在不同时间粒度上观察数据的能力（假表的详细介绍可参见：[分析教程-假表层函数](#)）。

表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

字段

序号	字段名	主键	数据来源	数据类型	字段描述
1	年	<input checked="" type="checkbox"/>		整数	

层函数

序号	名称	目标维字段名	函数

表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

字段

序号	字段名	主键	数据来源	数据类型	字段描述
1	月	<input checked="" type="checkbox"/>		整数	

层函数

序号	名称	目标维字段名	函数
1	年	年	int(?/100)

表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

字段

序号	字段名	主键	数据来源	数据类型	字段描述
1	日	<input checked="" type="checkbox"/>		日期	

层函数

序号	名称	目标维字段名	函数
1	月	月	year(?)*100+month(?)
2	年	年	year(?)



为订单表中的签单日期和发货日期字段添加外键。

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	客户
2	fk2	日
3	fk3	日

外键映射

序号	外键字段	引用字段
1	签单日期	日

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	客户
2	fk2	日
3	fk3	日

外键映射

序号	外键字段	引用字段
1	发货日期	日

为订单表中的  
签单日期和发  
货日期添加外  
键



参照上一节常规查询的如下步骤配置，进入多维分析页面：

启动DQL  
服务

配置数据  
源

Tag标签  
发布



通过在元数据将表看做维度、外键指向表的字段看做维度的属性，页面就可以通过树状目录的方式逐级展开。

按年月日层级来观察不同时期的客户签单情况：

报表名称

年:月	城市编码	大连		上海		客户名称 计数
		客户名称 计数	订单金额 求和	客户名称 计数	订单金额 求和	
2011	201102	2	1276	0	0	1
	201103	0	0	0	0	0
	201104	1	1420	0	0	0
	201105	2	5696	0	0	0
	201106	1	7390	0	0	0
	201107	3	1200	0	0	0
	201108	2	11595	1	1020	0
	201109	3	5126	0	0	2
	201110	0	0	0	0	0
	201111	2	372	0	0	0
	201112	0	0	0	0	2
	总计		16	34075	1	1020
201201	5	6518	0	0	0	

数据分析设置器

报表列表

添加报表

报表名称

- 订单
  - 订单ID
  - 客户ID
    - 客户名称
    - 联系人姓名
    - 联系人职务
  - 城市编码
    - 城市
    - 人口
- 雇员ID
  - 订单金额
- 签单日期
  - 月
  - 年
- 发货日期
  - 月
  - 年
- 客户
  - 客户ID
  - 客户名称
  - 联系人姓名
  - 联系人职务

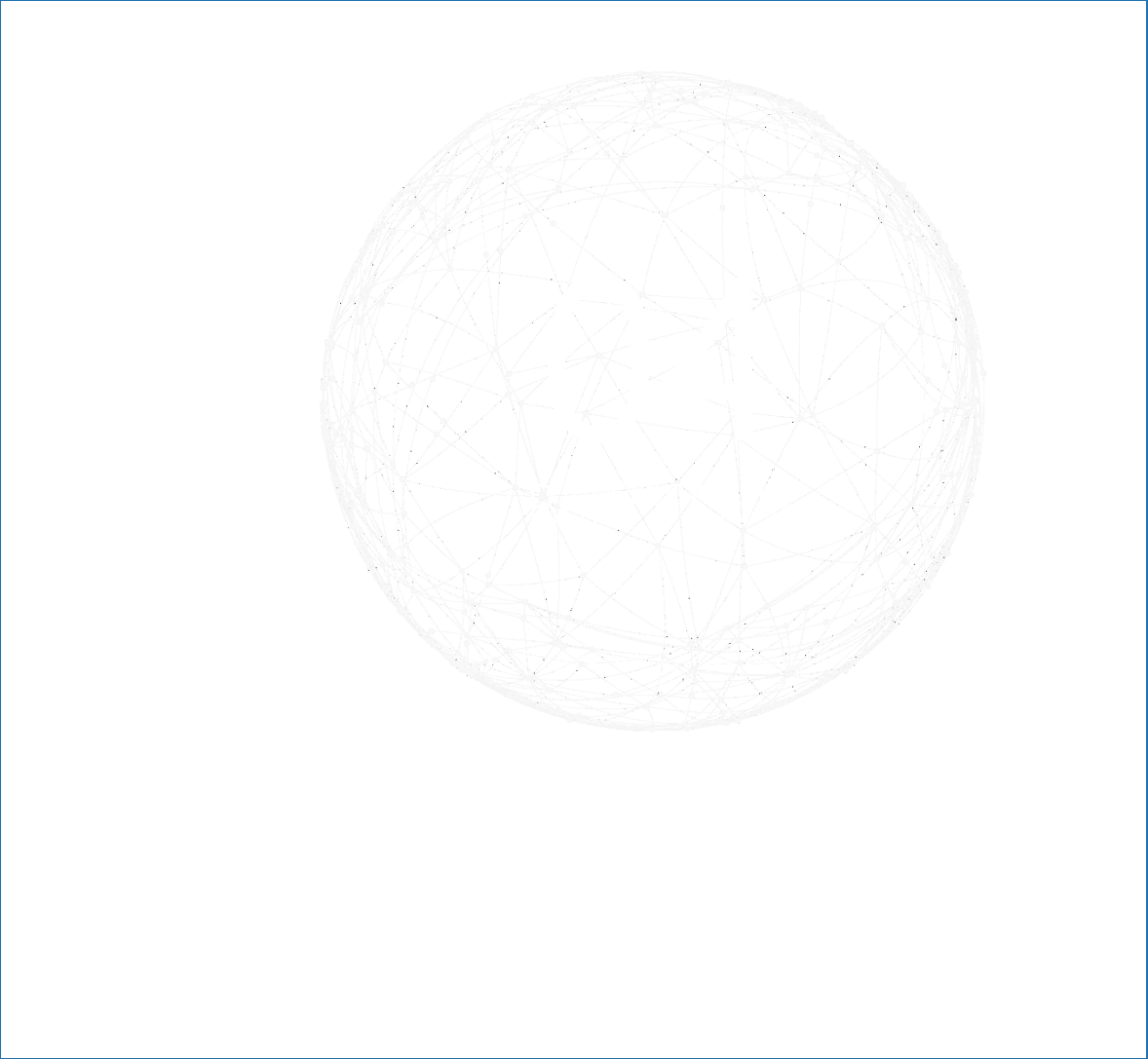
城市编码

指标标题

- 客户名称计数
- 订单金额求和

年

月







## 同维关联

如右图所示，雇员表中有一对同维字段，字段类型、长度、取值都相同，如果想按层级汇总需要关联多次，遇到这种情况常用的页面自动关联是无法处理的，润乾多维分析结合DQL服务提供了更有效的处理方式，下面来具体了解下。





打开前面做好的.1md元数据文件，导入雇员表，将雇员ID设置为主键：

导入数据库表

表名关键字

确定(O) 取消(C) 高级(A)

序号	名称	<input type="checkbox"/> 选出
7	日交易汇总	<input type="checkbox"/>
8	日市值	<input type="checkbox"/>
9	三板股票	<input type="checkbox"/>
10	月交易汇总	<input type="checkbox"/>
11	订单回款月销售汇总	<input type="checkbox"/>
12	订单明细	<input type="checkbox"/>
13	供应商	<input type="checkbox"/>
14	雇员	<input checked="" type="checkbox"/>
15	回款单	<input type="checkbox"/>

指定模式 DEMO

表名前加模式名(模式名.表名)

表名 雇员 物理表名 雇员

记录数 0 表描述

常规 外键 伪字段 基础表 汇总表 子表

字段  输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	雇员ID	<input checked="" type="checkbox"/>	雇员ID	整数	
2	姓名	<input type="checkbox"/>	姓名	字符串	
3	职务	<input type="checkbox"/>	职务	字符串	
4	出生日期	<input type="checkbox"/>	出生日期	日期	
5	雇佣日期	<input type="checkbox"/>	雇佣日期	日期	
6	上级	<input type="checkbox"/>	上级	整数	
7	籍贯城市编码	<input type="checkbox"/>	籍贯城市...	整数	



为订单表的**雇员ID**和雇员表的**上级**字段添加外键关联，雇员表的**上级**字段外键关联的表为雇员表本身：

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	客户
2	fk2	日
3	fk3	日
4	fk4	雇员

外键映射

序号	外键字段	引用字段
1	雇员ID	雇员ID

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	雇员

外键映射

序号	外键字段	引用字段
1	上级	雇员ID



参照常规查询小节的如下步骤配置，进入多维分析页面：

启动DQL  
服务

配置数据  
源

Tag标签  
发布



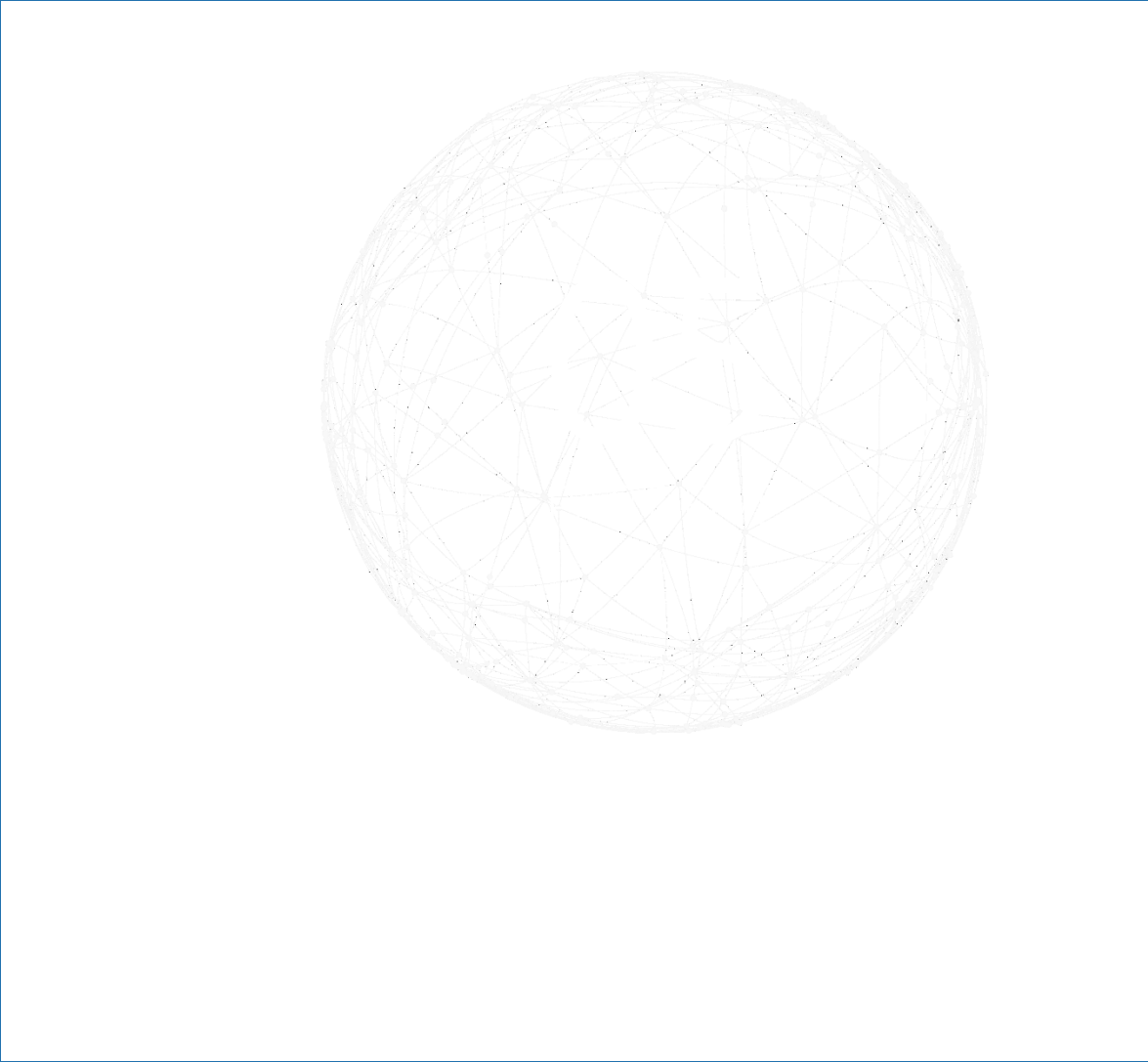
查询各区域部门领导的订单销售额汇总：

The screenshot displays a BI tool interface with two main panels. The left panel shows a report titled '报表名称' (Report Name) with a table of order sales data. The right panel shows the '数据分析设置器' (Data Analysis Settings) for the report, including a field list and a table configuration.

城市编码:上级		订单金额求和	总计
大连	郑建杰	56262	56262
	总计	56262	56262
上海	张雪眉	8701	8701
	总计	8701	8701
青岛	张雪眉	15381	15381
	总计	15381	15381
南京	刘英玫	127462	127462
	总计	127462	127462
常州	李芳	58486	58486
	总计	58486	58486
南昌	张雪眉	34098	34098
	总计	34098	34098
温州	张雪眉	52241	52241
	总计	52241	52241

The right panel shows the '订单' (Orders) table structure with fields like '订单ID', '客户ID', '客户名称', '联系人姓名', '联系人职务', '城市编码', '上级', '雇员ID', '姓名', '职务', '出生日期', '雇佣日期', '籍贯城市编码', '订单金额', and '签单日期'. The '城市编码' and '上级' fields are highlighted with red arrows pointing to the '城市编码' and '上级' dropdowns in the table configuration. The '订单金额' field is also highlighted with a red arrow pointing to the '订单金额求和' dropdown in the table configuration.

雇员表中雇员ID和上级为同维字段，在元数据中已将上级的外键关联字段指定为雇员表的雇员ID。在多维分析页面上会通过外键属性化将上级字段以树状的形式迭代显示，这样业务人员无需关心底层表结构关系，直接通过树状字段组织，拖拽所需指标即可。





很多业务系统的数据量巨大，直接从基础业务表进行查询汇总会非常慢，因此数据库里往往存在一些已经造好的历史汇总表。对于用户来说，当他进行查询时需要知道有哪些历史汇总表存在的难度比较大，尤其是当存在多级汇总表的时候，要能够恰当选择其中一个汇总表难度更大了。

**润乾多维分析为了减少业务人员对汇总表繁琐关系的学习，也为了提供更高效率的查询，DQL Server对汇总表提供了更优化的处理。**

**为汇总表设置基础表，当用户对基础表做分组汇总时，DQL语义层会智能寻找汇总表，看哪个汇总表和DQL的汇总级别最接近，然后智能的把DQL解析成从这个汇总表查出。**



## 汇总表

### 支付单

支付单ID
支付日期
支付金额
供应商ID
经办人
审核人

### 支付月汇总

年月
支付总额

### 支付供应商月汇总

年月
供应商
支付总额

数据库中有支付单、支付供应商月汇总、支付月汇总三张表。支付单是基础表，其他两张表是支付单的汇总表。基础表和汇总表需要在DQL Server元数据文件中定义关联关系。先对支付供应商月汇总、支付月汇总表分别添加支付单为基础表，添加完成后，查看支付单基础表时汇总表会自动列出。





打开前面做好的.1md元数据文件，导入供应商、支付单、支付供应商月汇总、支付月汇总：

导入数据库表

表名关键字

序号	名称	<input type="checkbox"/> 选出
22	销售	<input type="checkbox"/>
23	支付单	<input checked="" type="checkbox"/>
24	支付供应商汇总	<input type="checkbox"/>
25	支付供应商月汇总	<input checked="" type="checkbox"/>
26	支付月汇总	<input checked="" type="checkbox"/>
27	VIP客户	<input type="checkbox"/>
28	产品	<input type="checkbox"/>
29	订单	<input type="checkbox"/>
30	订单回款月汇总	<input type="checkbox"/>

指定模式

表名前加模式名(模式名.表名)

## 汇总表

## 制作元数据

为供应商、支付单、支付供应商月汇总、支付月汇总设置主键，支付供应商月汇总需要根据供应商和年月两个关键字段查找汇总表，因此需将年月和供应商均设置为主键：

表名 **供应商** 物理表名 供应商

记录数 0 表描述

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	供应商ID	<input checked="" type="checkbox"/>	供应商ID	整数	
2	供应商名称	<input type="checkbox"/>	供应商名称	字符串	
3	联系人姓名	<input type="checkbox"/>	联系人姓名	字符串	
4	联系人职务	<input type="checkbox"/>	联系人职务	字符串	
5	城市编码	<input type="checkbox"/>	城市编码	整数	

表名 **支付单** 物理表名 支付单

记录数 0 表描述

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	支付单ID	<input checked="" type="checkbox"/>	支付单ID	整数	
2	支付日期	<input type="checkbox"/>	支付日期	日期	
3	支付金额	<input type="checkbox"/>	支付金额	双精度数	
4	供应商ID	<input type="checkbox"/>	供应商ID	整数	
5	经办人	<input type="checkbox"/>	经办人	整数	
6	审核人	<input type="checkbox"/>	审核人	整数	

表名 **支付供应商月汇总** 物理表名 支付供应商月汇总

记录数 0 表描述

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	年月	<input checked="" type="checkbox"/>	年月	整数	
2	供应商	<input checked="" type="checkbox"/>	供应商	整数	
3	支付总额	<input type="checkbox"/>	支付总额	双精度数	

表名 **支付月汇总** 物理表名 支付月汇总

记录数 0 表描述

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	年月	<input checked="" type="checkbox"/>	年月	整数	
2	支付总额	<input type="checkbox"/>	支付总额	双精度数	

# 汇总表



# 制作元数据



为支付单、支付供应商月汇总、支付月汇总设置外键关联：

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	日
2	fk2	供应商
3	fk3	雇员
4	fk4	雇员

外键映射

序号	外键字段	引用字段
1	支付日期	日

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	月
2	fk2	供应商

外键映射

序号	外键字段	引用字段
1	年月	月

表名  物理表名

记录数  表描述

常规 外键 伪字段 基础表 汇总表 子表

外键

序号	外键名称	对应表名
1	fk1	月

外键映射

序号	外键字段	引用字段
1	年月	月



对支付月汇总和支付供应商月汇总表添加基础表设置，如下图所示字段名是“支付月汇总”和“支付供应商月汇总”表的字段名，广义外键或表达式是指基于支付单基础表的计算表达式：

表名  物理表名

记录数  表描述

常规 外键 伪字段 **基础表** 汇总表 子表

基础表

序号	基础表名
1	支付单

字段

序号	字段名	广义外键或表达式
1	年月	支付日期#月
2	支付总额	sum(支付金额)

表名  物理表名

记录数  表描述

常规 外键 伪字段 **基础表** 汇总表 子表

基础表

序号	基础表名
1	支付单

字段

序号	字段名	广义外键或表达式
1	年月	支付日期#月
2	供应商	供应商ID
3	支付总额	sum(支付金额)

- (1) 汇总表与基础表之间是一对多的关系；
- (2) 在定义基础表功能时，其中的字段名，必须包括主表的所有关键字段，如上图中的年月为支付月汇总主键，必须定义广义外键或表达式。计算字段可以全写，或只写一个。
- (3) 当基础表中有多个汇总表可用时，会通过表中记录数的设置选择记录数少的使用，记录数相同或都为空时取第一个符合条件的汇总表。



参照常规查询小节的如下步骤配置，进入多维分析页面：

启动DQL  
服务

配置数据  
源

Tag标签  
发布



对支付单按年月汇总支付金额：

报表名称

年,月	支付金额 求和	总计
201101	27445	27445
201107	6991	6991
201108	32553	32553
201109	31836	31836
201110	41932	41932
201111	50955	50955
201112	76748	76748
总计	268460	268460
201201	52055	52055
201202	71691	71691
201203	47104	47104
201204	59432	59432
201205	40457	40457
201206	70750	70750

数据分析设置器

报表列表

添加报表

报表名称

- 雇员
  - 雇员ID
  - 姓名
  - 职务
  - 出生日期
  - 雇佣日期
  - 上级
  - 籍贯城市编码
- 支付单
  - 支付单ID
  - 支付日期
    - 月
    - 年
  - 支付金额
- 供应商ID
- 经办人
- 审核人
- 供应商
  - 供应商ID
  - 供应商名称
  - 联系人姓名
  - 联系人职务

上表头

年 ▼

月 ▼

支付金额求和 ▼

+

系统自动根据月层级指标，查找到匹配的支付月汇总表，从已有的支付单月汇总表里查询数据进一步汇总。即执行以下SQL：**【SELECT CAST(T\_1\_1."年月"/100 AS INT) "年",T\_1\_1."年月" "月",T\_1\_1."支付总额" "支付金额求和" FROM "支付月汇总" T\_1\_1】**



对支付单按供应商和月汇总支付金额：

报表名称

月	201101	201107	201108	201109	201110	201111
供应商ID	支付金额求和	支付金额求和	支付金额求和	支付金额求和	支付金额求和	支付金额求和
1	1938	0	1330	2099	1120	1064
2	39	0	2381	427	1142	3196
3	1085	0	450	0	0	3700
4	31	0	570	1209	0	0
5	286	0	456	0	252	1050
6	752	209	0	490	0	591
7	1451	0	2441	3689	7493	4914
8	403	3240	786	1833	614	492
9	18	0	126	0	2312	4110
10	135	0	248	27	99	0
11	0	0	1098	658	0	2508
12	9278	0	1589	3244	5548	2096
13	104	0	1553	155	259	1501

数据分析设置器

报表列表 + 添加报表

报表名称 打印 删除

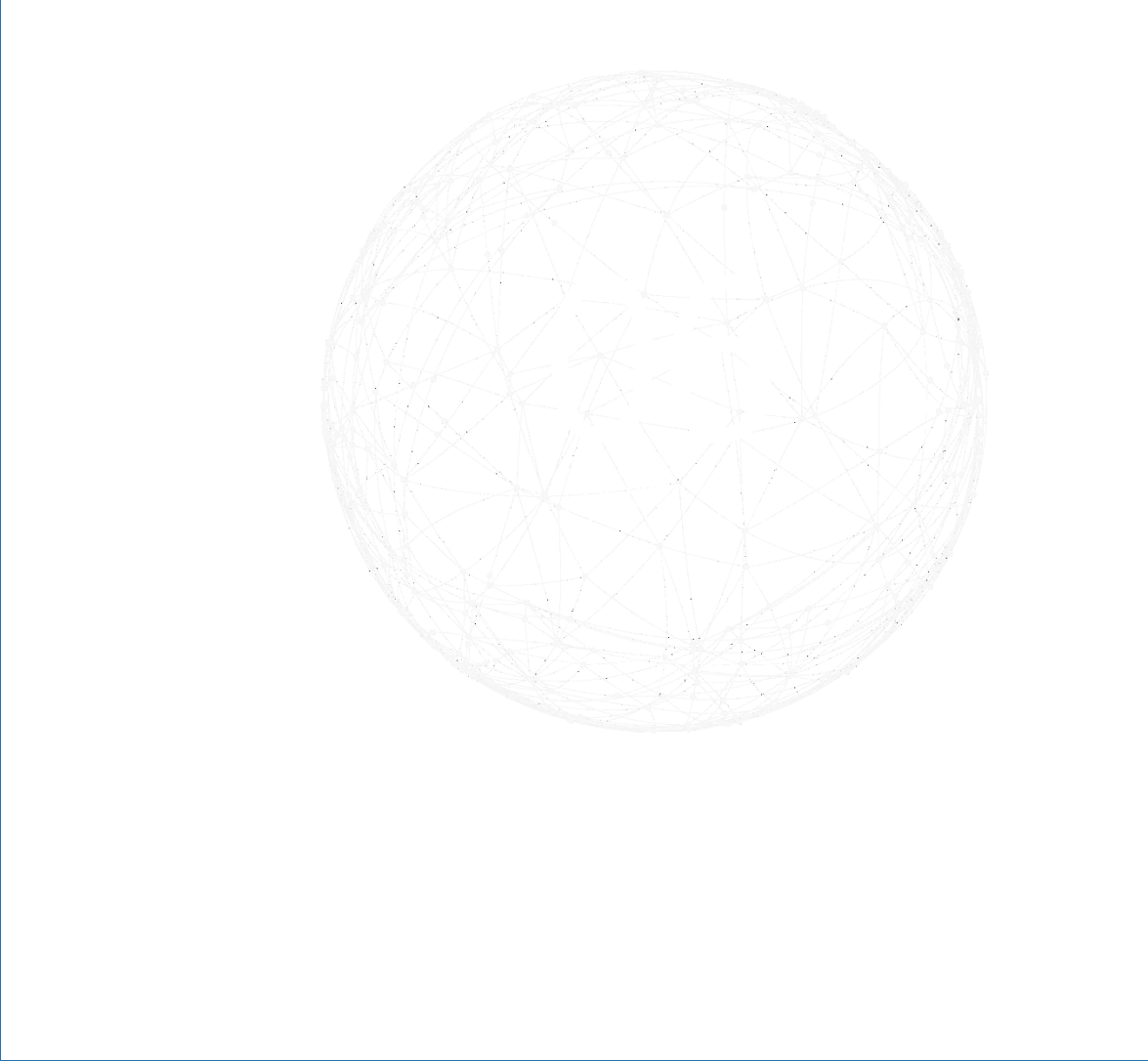
- [-] 雇员
  - [-] 雇员ID
  - [-] 姓名
  - [-] 职务
  - [-] 出生日期
  - [-] 雇佣日期
  - [-] 上级
  - [-] 籍贯城市编码
- [-] 支付单
  - [-] 支付单ID
  - [-] 支付日期
    - [-] 月
    - [-] 年
  - [-] 支付金额
  - [-] 供应商ID
  - [-] 经办人
  - [-] 审核人
- [-] 供应商
  - [-] 供应商ID
  - [-] 供应商名称
  - [-] 联系人姓名
  - [-] 联系人职务
  - [-] 城市编码

月

供应商ID

支付金额求和

当聚合指标为供应商和月时，系统自动匹配到支付供应商月汇总表，从已有的支付供应商月汇总表里查询数据。即执行以下SQL：**【SELECT T\_1\_1."年月" "月",T\_1\_1."供应商" "供应商ID",T\_1\_1."支付总额" "支付金额求和" FROM "支付供应商月汇总" T\_1\_1】**，通过对基础表的设定，可以充分利用事先汇总完成的数据，以提高效率。







## 大数据跨库分析

Mysql数据库记录的华北地区的回款单表结构：

回款单	客户
回款日期	客户编号
客户编号	客户名称
回款金额	联系人姓名
销售编号	联系人职务
合同编号	城市编码

Hsql数据库记录的华南地区的回款单表结构：

回款单	客户
回款日期	客户ID
客户编号	客户名称
回款金额	联系人姓名
销售编号	联系人职务
合同编号	城市编码



针对左侧这种数据分  
别来自不同的数据库，  
并且数据库中的表结  
构也不同，应该怎样  
做跨库分析呢？



分别创建使用Mysql和HSQL数据源的元数据文件，需保证元数据中做混合查询的表结构一致：

Mysql对应的元数据文件mysqldemo.lmd:

表名: 回款单 物理表名: 回款单  
记录数: 0 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	回款单ID	<input checked="" type="checkbox"/>	回款单编号	整数	
2	回款日期	<input type="checkbox"/>	回款日期	日期	
3	客户ID	<input type="checkbox"/>	客户ID	字符串	
4	金额	<input type="checkbox"/>	金额	双精度数	
5	销售ID	<input type="checkbox"/>	销售编号	整数	

表名: 客户 物理表名: 客户  
记录数: 10 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	客户	<input checked="" type="checkbox"/>	客户编号	字符串	
2	客户名称	<input type="checkbox"/>	客户名称	字符串	
3	联系人姓名	<input type="checkbox"/>	联系人姓名	字符串	
4	联系人职务	<input type="checkbox"/>	联系人职务	字符串	
5	市	<input type="checkbox"/>	城市编码	整数	

Hsql对应的元数据文件hsqldemo.lmd:

表名: 回款单 物理表名: 回款单  
记录数: 0 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	回款单ID	<input checked="" type="checkbox"/>	回款单ID	整数	
2	回款日期	<input type="checkbox"/>	回款日期	日期	
3	客户ID	<input type="checkbox"/>	客户ID	字符串	
4	金额	<input type="checkbox"/>	金额	双精度数	
5	销售ID	<input type="checkbox"/>	销售ID	整数	

表名: 客户 物理表名: 客户  
记录数: 10 表描述:

常规 外键 伪字段 基础表 汇总表 子表

字段 输入字段名关键字进行查找:

序号	字段名	主键	数据来源	数据类型	字段描述
1	客户	<input checked="" type="checkbox"/>	客户ID	字符串	
2	客户名称	<input type="checkbox"/>	客户名称	字符串	
3	联系人姓名	<input type="checkbox"/>	联系人姓名	字符串	
4	联系人职务	<input type="checkbox"/>	联系人职务	字符串	
5	市	<input type="checkbox"/>	城市编码	整数	



为两个元数据文件的回款单分别添加外键，将客户ID与客户表关联：

表名  物理表名

记录数  表描述

外键

序号	外键名称	对应表名
1	fk1	客户

外键映射

序号	外键字段	引用字段
1	客户ID	客户



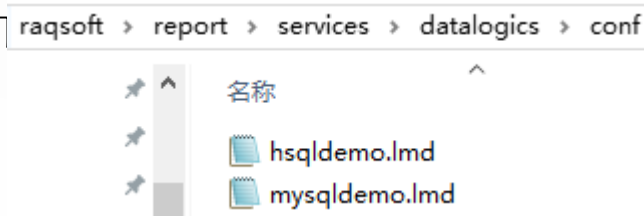
在[WEB应用根目录]/WEB-INF/raqsoftConfig.xml中添加元数据层用到的Mysql和HSq1物理数据源

```
<?xml version="1.0" encoding="UTF-8"?><Config Version="1"><Runtime >
//配置应用需要用的数据源
<DBList>
<!--元数据使用的物理数据源-->
<!--数据库类型编号，编号依次代表的数据库类型为：[UNKNOWN:0; ORACLE:1; SQLSVR:2; SYBASE:3; SQLANY:4; INFMIX:5;
FOXPRO:6; ACCESS:7; FOXBAS:8; DB2:9; MYSQL:10; KINGBASE:11; DERBY:12; HSQ1:13; TERADATA:14; POSTGRES:15;
DATALOGIC:16; IMPALA:17; ESSBASE:101;] -->
<DB name="mysqldb">
  <property name="url" value="jdbc:mysql://192.168.0.123:3306/dqldb?useCursorFetch=true" />
  <property name="driver" value="com.mysql.jdbc.Driver" />
  <property name="type" value="10" />
  <property name="user" value="root" />
  <property name="password" value="root"/>
  ... ..
</DB>
  <DB name="hsqldb">
  <property name="url" value="jdbc:hsqldb:hsqldb://127.0.0.1/dqldemo" />
  <property name="driver" value="org.hsqldb.jdbcDriver" />
  <property name="type" value="13" />
  <property name="user" value="sa" />
  <property name="password" />
  ... ..
</DB>
</DBList> ... ..
</Config>
```



将语义层文件和数据源信息部署至DQL服务：

(1) 将mysqldemo.lmd和hsqldemo.lmd文件放至[DQL Server服务目录]/conf



(2) 在[DQL Server服务目录]下的service.xml中添加<MAP/>标签

<!-- DQL Server服务使用多物理库时的文件结构配置，ENTRY中未配置元数据层文件时则会采用SERVICE标签中的同名属性值。

dbName:为raqsoftConfig.xml中的物理数据源名称； dictionary: DQL Server服务的元数据层字典文件(.dct)，不是必填项，可选； logicmetadata: DQL Server服务的元数据文件(.lmd)，不是必填项，可选

```
<MAP>      <ENTRY dbName="mysql" logicmetadata="mysqldemo.lmd" />
            <ENTRY dbName="hsqldb" logicmetadata="hsqldemo.lmd"/></MAP>
```

(3) 在server.xml中添加raqsoftConfig和mdb属性

<!--raqsoftConfig: 报表配置文件raqsoftConfig.xml路径，此属性仅在使用多个物理库时有效 -->

```
<SERVER host="127.0.0.1" port="3366" timeout="30000" log="true" license="" raqsoftConfig="web/webapps/demo/WEB-INF/raqsoftConfig.xml" >
```

<!-- SERVICE:要启动的DQL Server列表，可以配置多个，多个时则需要添加多条<service/>，每个<service/>的name属性所指定的服务都必须位于[应用根目录]services下，services目录下可以有多个DQL Server服务目录，但只有下面列出的服务才会被启动 -->

<!-- name: DQL Server的名称，且名称必须与文件夹名称一致 -->

<!-- mdb:是否使用多个物理库，缺省为false。为true时表示使用多个物理库-->

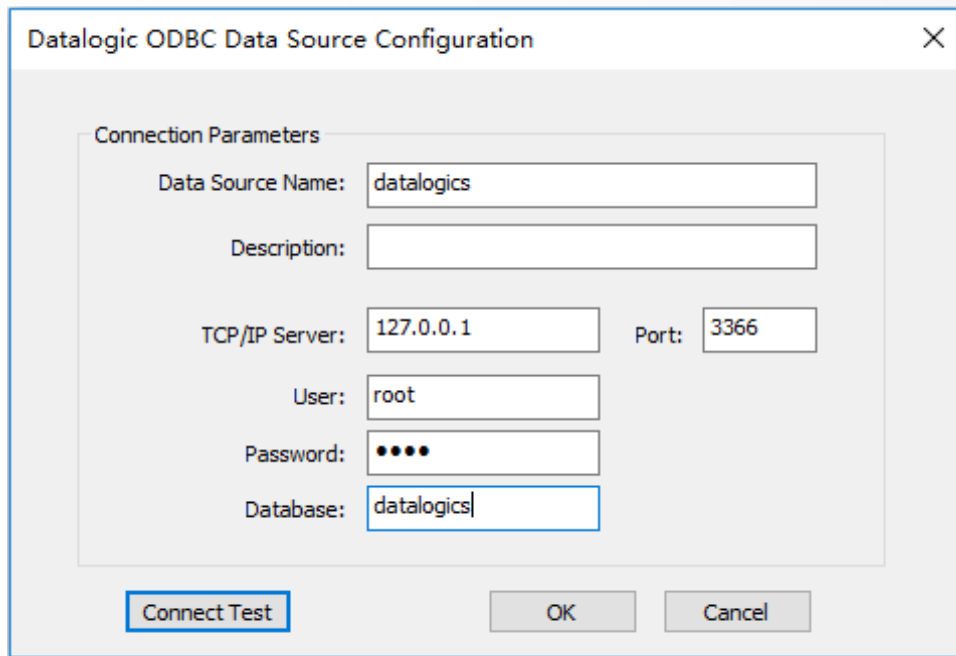
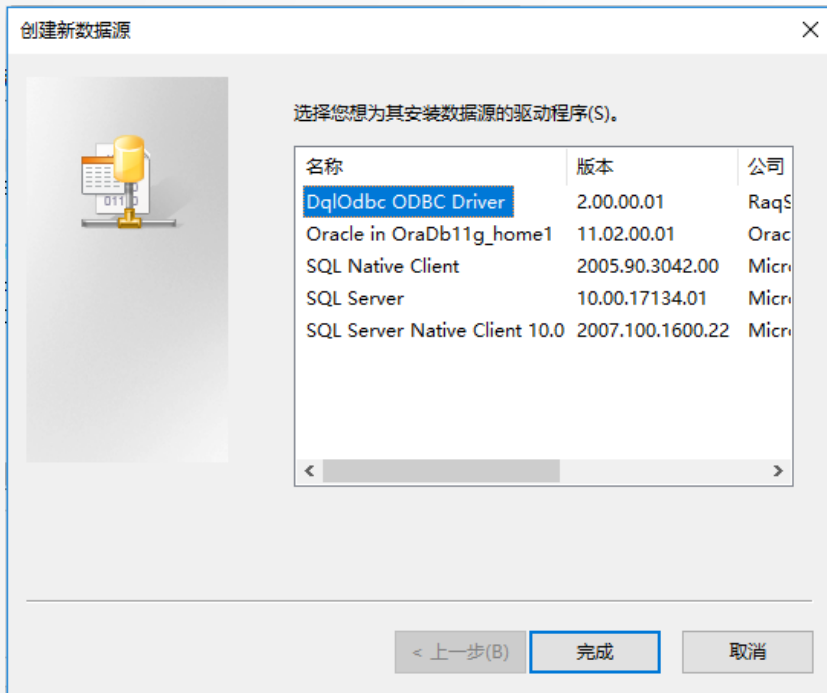
```
<SERVICE name="datalogics" mdb="true"/>
```

```
</SERVER>
```

(4) 启动DQL服务



在ODBC数据源管理程序的“用户DSN”栏下，选择“DQL ODBC Driver” 点击添加，出现创建新数据源界面：



具体操作步骤和属性含义可参考《[分析教程](#)》[使用 ODBC 连接 DQL Server](#)章节。



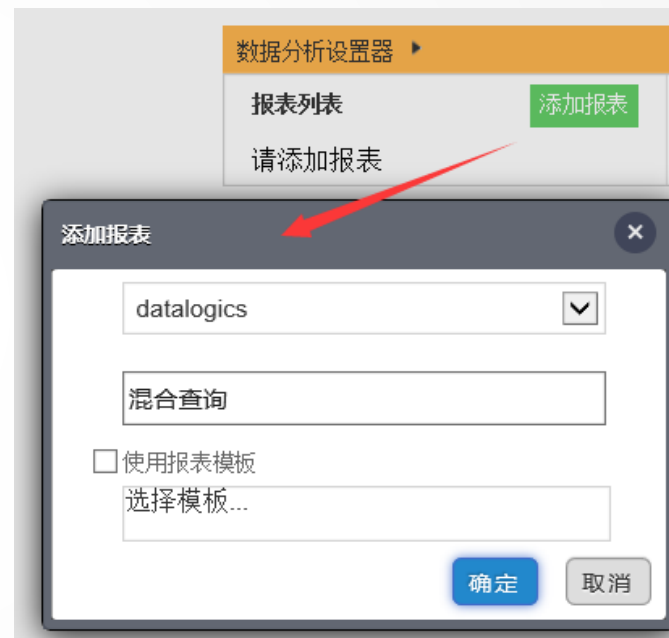
在[WEB应用根目录]/WEB-INF/raqsoftConfig.xml中添加DQL ODBC数据源信息

```
<?xml version="1.0" encoding="UTF-8"?><Config Version="1"><Runtime >
//配置应用需要用的数据源
<DBList>
<!--元数据使用的物理数据源-->
<!--数据库类型编号，编号依次代表的数据库类型为：[UNKNOWN:0; ORACLE:1; SQLSVR:2; SYBASE:3; SQLANY:4; INFMIX:5;
FOXPRO:6; ACCESS:7; FOXBAS:8; DB2:9; MYSQL:10; KINGBASE:11; DERBY:12; HSQL:13; TERADATA:14; POSTGRES:15;
DATALOGIC:16; IMPALA:17; ESSBASE:101;] -->
<DB name="DataLogics">
  <property name="url" value="jdbc:odbc:datalogics" />
  <property name="driver" value="sun.jdbc.odbc.JdbcOdbcDriver" />
  <property name="type" value="0" />
  <property name="user" value="root" />
  <property name="password" value="root" />
  <property name="batchSize" value="1000" />
  <property name="autoConnect" value="false" />
  <property name="useSchema" value="false" />
  <property name="addTilde" value="false" />
  <property name="dbCharset" value="GBK" />
  <property name="clientCharset" value="GBK" />
  <property name="needTransContent" value="false" />
  <property name="needTransSentence" value="false" />
  <property name="caseSentence" value="false" />
</DB>
</DBList> ... ..</Config>
```



启动Tomcat，访问多维分析页面，进行混合查询分析：

url: <http://localhost:6868/demo/raqsoft/guide/jsp/olap.jsp>



分析控件中新建数据集，选择使用多物理库DQL Server服务的数据源Datalogics，编写查询语句，设置查询语句后，点击“查询数据，缓存入文件”，最后保存。然后添加报表。





查询回款单，展现来自异构数据库的混合数据：

The screenshot shows a software interface for data analysis. On the left, a table titled '混合查询' (Mixed Query) displays data from two different sources. On the right, a '数据分析设置器' (Data Analysis Configurator) panel shows the configuration for the query, including a list of fields and a complex condition section.

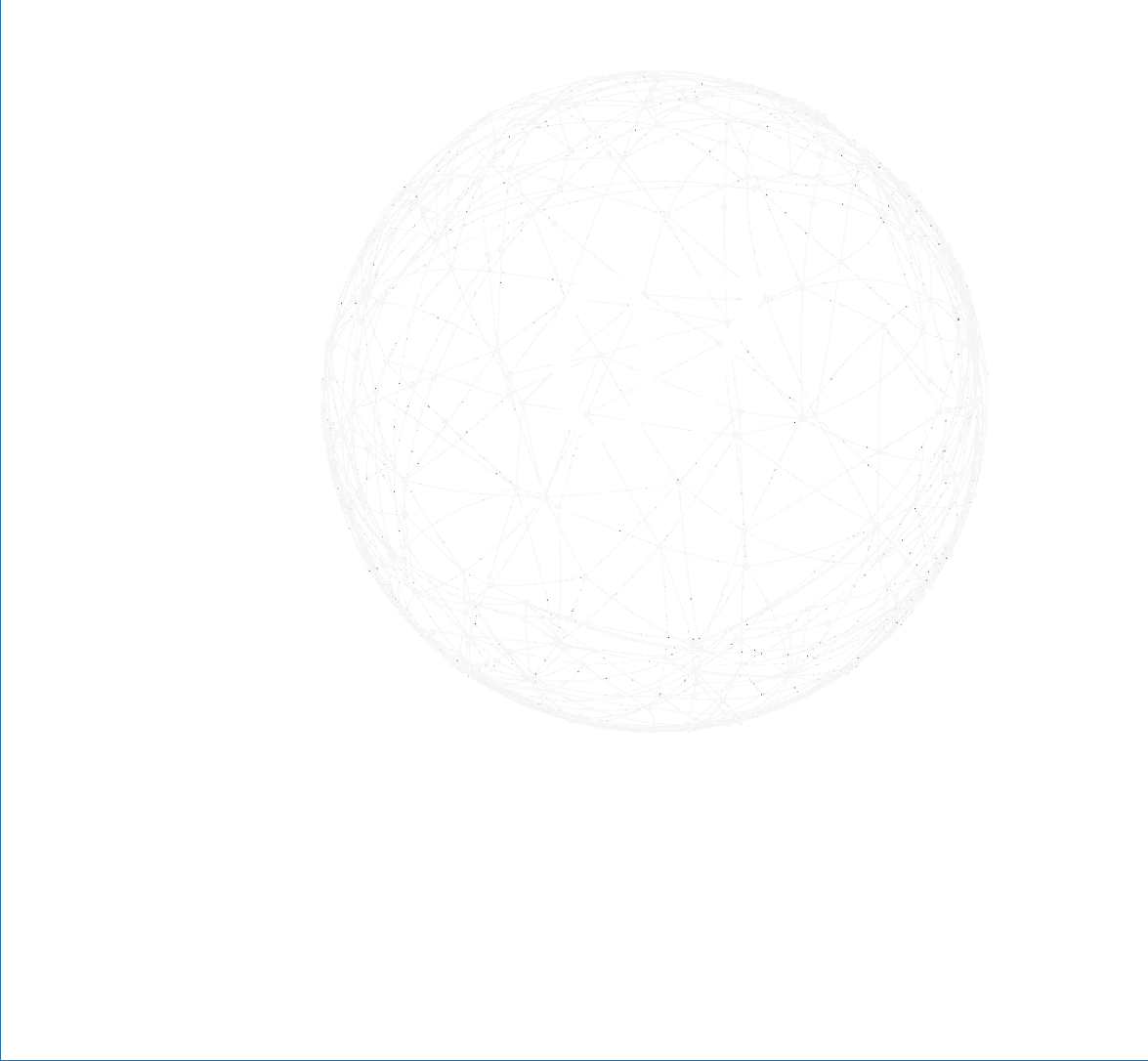
回款单ID	客户名称	金额	销售ID
1	山泰企业	20	2
2	东帝望	112	9
3	实翼	29	9
4	千固	270	9
5	福星制衣厂股份有限公司	1586	2
393	凯旋科技	509	2
394	远东开发	1658	7
395	百达电子	343	2
396	迈多贸易	747	9
397	凯威国际顾问公司	1418	7
398	实翼	409	9
399	师大贸易	1042	2
400	兰格英语	92	1

Hsql数据库回款单表记录：

回款单ID	回款日期	客户ID	金额	销售ID
1	2011-9-1	VINET	20	2
2	2011-7-29	TOMSP	112	9
3	2011-7-19	HANAR	29	9
4	2011-9-9	VICTE	270	9
5	2011-10-7	SUPRD	1,586	2

Mysql数据库回款单表记录：

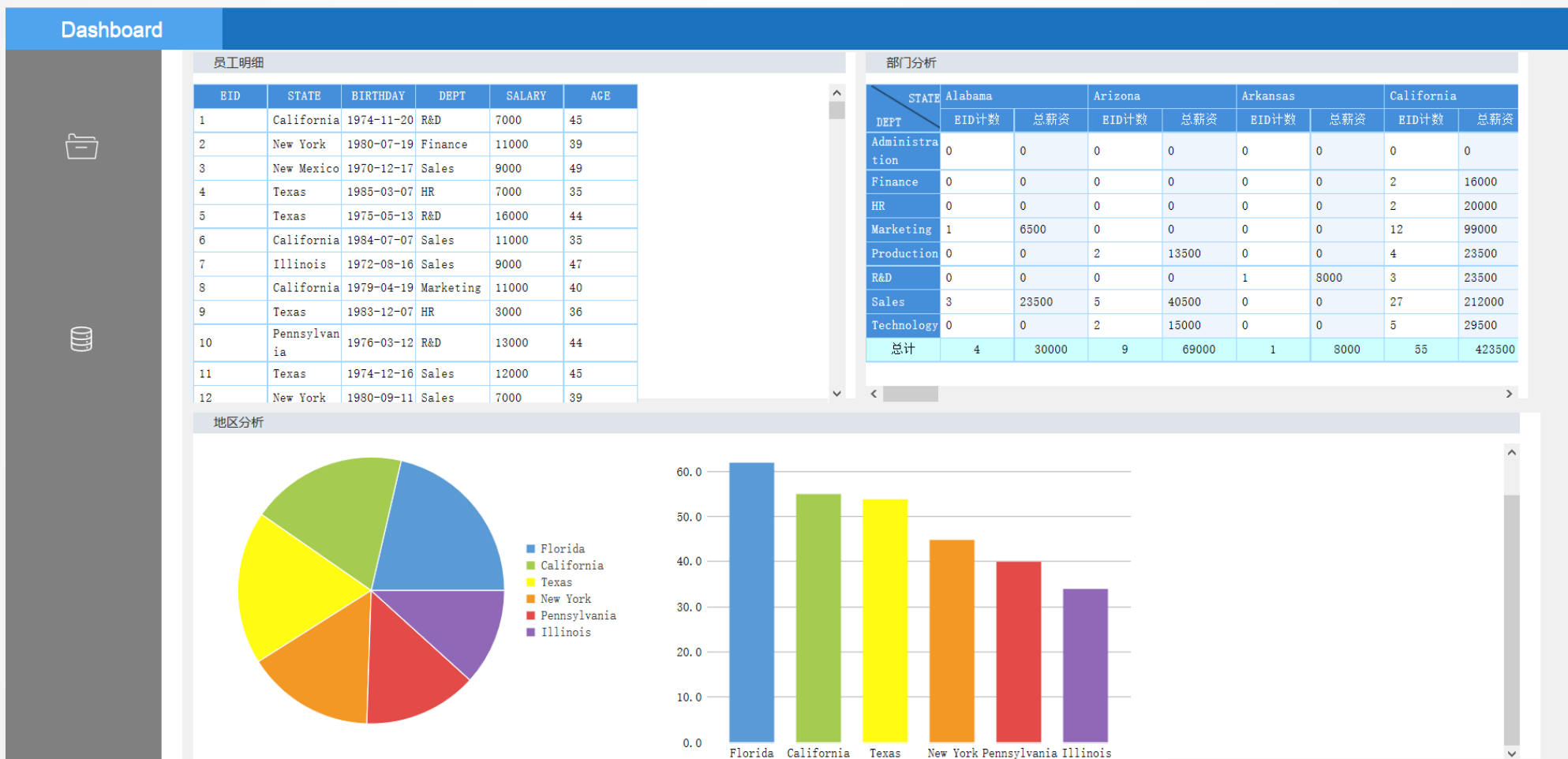
回款单编号	回款日期	客户编号	回款金额	销售编号
393	2012-10-22	WANDK	509	2
394	2012-11-24	HILAA	1,658	7
395	2012-9-20	SIMOB	343	2
396	2012-9-18	BOLID	747	9
397	2012-9-3	WELLI	1,418	7
398	2012-11-29	HANAR	409	9
399	2012-10-8	HUNGO	1,042	2
400	2012-10-12	QUEDE	92	1





## Dashboard分析

前面我们体验到了可视化拖拽模式对关联分析的灵活处理。除了多维分析外，润乾报表还有Dashboard分析界面—仪表盘，简称DBD，即大屏和分析控件的结合，同样也能对关联分析灵活展现。





## Dashboard分析

在浏览器中输入访问地址：

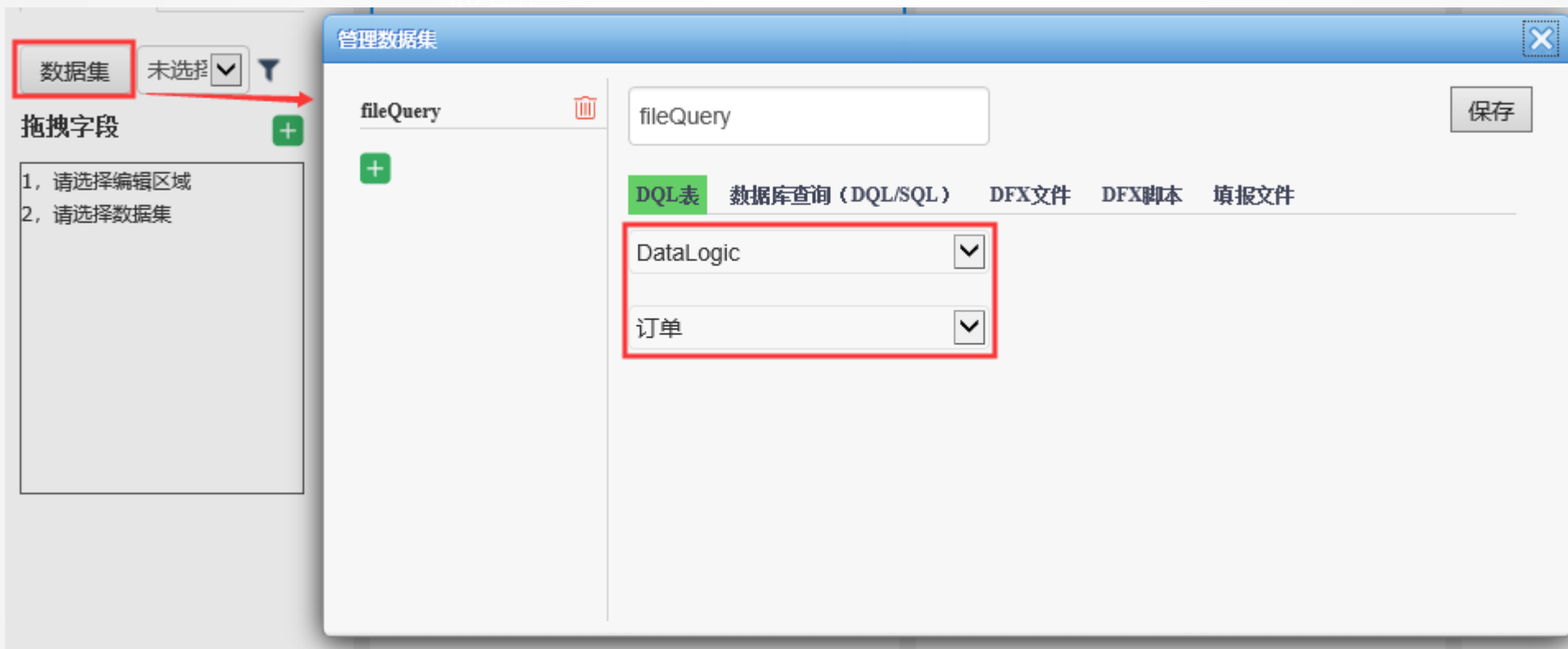
<http://localhost:6868/demo/raqsoft/guide/dbd/jsp/index.jsp>

The screenshot displays the Dashboard software interface. At the top, there is a blue header bar with the text "Dashboard". Below the header, there is a navigation bar with buttons for "保存" (Save), "打开" (Open), "预览" (Preview), "选择布局" (Select Layout), and "共享参数" (Share Parameters). The main workspace is divided into a left sidebar and a central grid area. The sidebar contains a folder icon, a "数据源" (Data Source) button, a dropdown menu showing "未选择" (Not Selected), and a "拖拽字段" (Drag Fields) section with a green plus sign and a list of instructions: "1, 请选择编辑区域" (1, Please select the edit area) and "2, 请选择数据集" (2, Please select the data set). The central grid area consists of six empty chart slots arranged in a 2x3 grid, labeled "图表名称0" through "图表名称5". Each slot has a header bar with its name and icons for zooming and deleting.



## Dashboard分析

点击数据集按钮，在弹出的管理数据集窗口编辑数据集名称和数据集内容，如下图：





## Dashboard分析

选中报表块，左侧选择数据集，拖拽字段到指标区，按年月层级来观察不同时期的客户签单情况：

The screenshot displays a dashboard configuration interface. On the left, there is a sidebar with a folder icon and a database icon. The main area is divided into several sections:

- Top Bar:** Contains buttons for '保存' (Save), '打开' (Open), '预览' (Preview), '选择布局' (Select Layout), and '共享参数' (Share Parameters).
- Chart Type:** A tab labeled '统计图' (Statistical Chart) is selected.
- Data Source:** A dropdown menu shows 'DQL' as the selected data source.
- Field Selection:** A section titled '拖拽字段' (Drag Fields) contains a list of fields from a table named '订单' (Orders):
  - 订单ID
  - 客户ID
  - 雇员ID
  - 签单日期
  - 发货日期
  - 订单金额
- Configuration Panel:** A panel below the field selection shows a hierarchy for '城市编码' (City Code):
  - 年 (Year)
  - 月 (Month)Below this, there are two indicator titles: '客户名称计数' (Customer Name Count) and '订单金额计数' (Order Amount Count).
- Data Table:** A table titled '图表名称0' (Chart Name 0) displays data for the year 2011, grouped by city code. The table has columns for '年:月' (Year:Month), '城市编码' (City Code), '客户名称计数' (Customer Name Count), and '订单金额计数' (Order Amount Count). The data is split into two sub-groups: '10101' and '20101'.

年:月	城市编码 10101		城市编码 20101		
	客户名称计数	订单金额计数	客户名称计数	订单金额计数	
2011	201102	2	2	0	0
	201103	0	0	0	0
	201104	1	1	0	0
	201105	2	2	0	0
	201106	1	1	0	0
	201107	3	3	0	0
	201108	2	2	1	1
	201109	3	3	0	0
	201110	0	0	0	0
	201111	2	2	0	0



## Dashboard分析

报表块不仅能通过普通报表使数据一目了然，还可以使用统计图的辅助使得报表更直观。如何使用条形图展现雇员每年的签单情况呢？

Dashboard

保存 打开 预览 选择布局 共享参数

网格报表 统计图

数据集 DQL 拖拽字段

订单

- 订单ID
- 客户ID
- 雇员ID
- 签单日期
- 发货日期
- 订单金额

条形图

分类	年
系列	雇员ID
系列值	订单金额

图表名称0

年:月	计数	计数	计数	计数
201102	2	2	0	0
201103	0	0	0	0
201104	1	1	0	0
201105	2	2	0	0
201106	1	1	0	0
201107	3	3	0	0
201108	2	2	1	1
201109	3	3	0	0
201110	0	0	0	0
201111	2	2	0	0
201112	0	0	0	0
总计	16	16	1	1

图表名称1

图表名称3

图表名称4

更多关于Dashboard页面的使用可参考《[润乾报表 dashboard 分析](#)》。

想要了解更多 请联系我们



THANK YOU